CLAIMS

1. A method of scheduling the handling of data from a plurality of channels, comprising:

accumulating data from a plurality of channels by a remote access server;

scheduling a processor of the server to handle the accumulated data from at least one first one of the channels, once during a first cycle time; and

scheduling the processor to handle the accumulated data from at least one second one of the channels, once during a second cycle time different from the first cycle time.

2. A method according to claim 1, wherein the first cycle begins concurrently with a second cycle.

3. A method according to claim 2, wherein the first cycle time is an integer multiple of the second cycle time.

4. A method according to claim 1, wherein scheduling the processor to handle the accumulated data comprises scheduling the processor, during the second cycle, to handle the accumulated data from substantially all the at least one second channels, before scheduling the processor to handle data from any other of the plurality of channels.

5. A method according to claim 4, wherein scheduling the processor to handle the accumulated data from the at least one first one of the channels comprises checking whether the second cycle has elapsed and scheduling the processor to handle the accumulated data from one of the at least one first channels only if the second cycle has not elapsed.

6. A method according to claim 1, wherein the at least one first one of the channels comprises a plurality of first channels and the at least one second one of the channels comprises a plurality of second channels.

7. A method according to claim 6, wherein the scheduling comprises scheduling the processor to handle the accumulated data from at least one of the second channels at least twice before scheduling the processor to handle data from at least one of the first channels.

8. A method according to claim 1, wherein scheduling the processor to handle the accumulated data comprises allowing the processor to utilize up to a predetermined amount of processing time for each channel.

9. A method according to claim 1, wherein the processor does not run an operating system which performs preemption.

10. A method according to claim 1, wherein scheduling the processor comprises having the processor wait without handling data from any of the channels if all the channels were scheduled for handling during their respective current cycles.

11. A method according to claim 10, comprising measuring the waiting time of the processor in the first cycle and using the measured time in determining whether to accept handling data from an additional channel.

12. A method according to claim 1, wherein the scheduling of handling the data of one channel is performed without interrupting the processor in the middle of handling accumulated data from a different channel.

13. A method according to claim 1, comprising processing an entire block of accumulated data of the scheduled channel responsive to the scheduling.

14. A method of scheduling the handling of a plurality of connections, comprising:

accumulating data from a plurality of channels by a remote access server which includes a processor which always processes data of a channel without interruption responsive to scheduling;

scheduling the processor to process data from a first one of the channels at least twice, with a first interval between the schedulings; and

scheduling the processor to process data from a second one of the channels at least twice, with a second interval between the schedulings, which second interval includes the entire first interval.

15. A method according to claim 14, wherein scheduling the processor comprises scheduling the processor to handle data from the first one of the channels once during a first cycle time and scheduling the processor to handle data from the second one of the channels once during a second cycle time longer than the first cycle time.

16. A method according to claim 14, comprising processing an entire block of accumulated data of the scheduled channel responsive to the scheduling.

17. A remote access server, comprising:

a plurality of channel drivers which accumulate data from respective channels;

a processor which handles the accumulated data; and

a scheduler which schedules the processor to handle accumulated data from a first channel once during a first cycle time and data from a second channel once during a second cycle time different from the first cycle time, without interrupting the processor while it is processing data from a channel.

18. A server according to claim 17, wherein the scheduler schedules the processor to handle the data from the first channel at least twice before scheduling the processor to handle data from the second channel.

19. A method of determining, by a remote access server, whether to accept an incoming connection, comprising:

determining an amount of unused processing time of a processor of the server; and

determining whether the amount of unused processing time is sufficient to handle the incoming connection.

20. A method according to claim 19, wherein determining the amount of unused processing time comprises determining by the processor.

21. A method according to claim 19, wherein determining the amount of unused processing time comprises measuring the amount of time in which the processor does not process data from any connection.

22. A method according to claim 19, wherein determining the amount of unused processing time comprises estimating the amount of time based on a number of connections being handled by the server.

23. A method according to claim 22, wherein estimating the amount of time comprises estimating based on the types of the connections being handled by the server.

24. A method according to claim 19, wherein determining whether the amount of unused processing time is sufficient to handle the incoming connection comprises determining whether the amount of unused processing time exceeds an amount sufficient to handle the incoming connection at least by a predetermined safety margin.

25. A method according to claim 24, wherein the safety margin has a size determined responsive to a number of connections being handled by the server.

26. A method of scheduling the handling of data, by a remote access server keeping track of a short cycle and a long cycle, from a plurality of channels including at least one short cycle channel and at least one long cycle channel, comprising:

accumulating data from the plurality of channels by the server;

scheduling a processor of the server to handle the accumulated data from all the short cycle channels;

determining whether a current short cycle has elapsed after scheduling the processor to handle the data from all the short cycle channels; and

scheduling the processor to handle the accumulated data from one of the at least one long cycle channel if the current short cycle did not elapse, if there is a long cycle channel which was not scheduled yet during the current long cycle.

27. A method according to claim 26, comprising determining whether the current short cycle has elapsed after scheduling the processor to handle the data from the long cycle channel, and scheduling the processor to handle the accumulated data from an additional long cycle channel, if the current short cycle did not elapse.

28. A method according to claim 26, comprising waiting, after scheduling the processor to handle the data from all the short cycle channels, until the beginning of the next short cycle without processing data from any channel, if all the long cycle channels were already scheduled during the current long cycle.

29. A method according to claim 26, wherein the long cycle begins concurrently with a short cycle.

30. A method according to claim 26, wherein the long cycle time is an integer multiple of the short cycle time.

31. A method of scheduling the handling of a plurality of connections, comprising:

accumulating data from a plurality of channels by a remote access server;

determining for at least one of the connections a quality of service level; and

scheduling the processor to process data from the plurality of connections in an order determined responsive to the determined quality of service level.

32. A method according to claim 31, wherein the scheduling comprises scheduling the processor to handle data from at least one first connection before handling data from at least one second connection having a lower quality of service level than the at least one first connection.

33. A method according to claim 31, comprising changing the quality of service level of at least one of the connections while accumulating the data and changing the order of scheduling responsive to the change in the quality of service level.